REMARKS/ARGUMENTS

Favorable reconsideration of this application, in light of the present amendments and following discussion, is respectfully requested.

Claims 1-20 are pending in the present application. Claims 1, 13 and 20 are amended by the present amendment. Support for the amended claims can be found in the original specification, claims and drawings.[1] No new matter is presented.

In the Office Action, Claims 1-10, 13-17 and 20 are rejected under 35 U.S.C. § 102(b) as anticipated by Loveman ("The DEC High Performance Fortran 90 Compiler Front End", IEEE 1995, pp. 46-53); and Claims 11-12 and 18-19 are rejected under 35 U.S.C. § 103(a) as unpatentable over Loveman.

In response to the above noted rejection under 35 U.S.C. § 102, Applicants respectfully submit that amended independent Claims 1, 13 and 20 recite novel features clearly not disclosed by Loveman.

Amended independent Claim 1, for example, recites, in part, a ... method of compiling for generating object code from an input source program, the object code including user-defined machine instructions defined by a user, the method comprising:

> ... replacing, by a code optimizer, the multiple machine instructions by a user-defined machine instruction stored in the intrinsic function definition database in the case where the multiple machine instructions generated from the input source program by the code generator are determined to exactly match the machine instructions generated from the details of an intrinsic function definition body of the user-defined machine instruction...

Independent Claims 13 and 20, while directed to alternative embodiments, are amended to recite similar features. Accordingly, the remarks and arguments presented below are applicable to each of amended independent Claims 1, 13 and 20.

[1] E.g., specification, at least at p. 19, l. 10 – p. 21, l. 26 and Fig. 10.

As disclosed in an exemplary embodiment at Fig. 10 and p. 19, l. 10 – p. 21, l. 26 of the specification, when the general (multiple) machine instructions generated from the input source program by the code generator exactly match the machine instructions generated from the details of the intrinsic function definition body of the user-defined machine instruction (the code optimizer determines whether there is a match (see Fig. 10)), the code optimizer replaces the multiple machine instructions by the user-defined machine instruction stored in the intrinsic function definition database.

Although the compiler of the claimed configuration can input the definition (operation definition of the associated instruction) of the intrinsic function, only the prototype declaration of the intrinsic function is used in Loveman. Further, in the claimed configuration, analyzing and retrieval are performed for the conventional program and the code generation of the instruction associated with the intrinsic function is performed. On the other hand, in Loveman, the code generation of the associated instruction is performed only from the description which explicitly called the intrinsic function.

Therefore, the claimed present invention differs from Loveman in that in the claimed configuration it is not necessary to rewrite the user application itself; a user does not need to call the intrinsic function explicitly within the user application; and it is not necessary to rebuild the compiler.

For example, in Loveman, when code generation is performed using an instruction peculiar to a target which executes this program to improve the performance of the program, the intrinsic function related with a special instruction, such as "pi=sum (rectangle_area)" is explicitly called. According to Loveman, therefore, it is necessary to rewrite the program directly, and when there are also many parts needed to improve performance of the program, it is also necessary for those programs to be rewritten. Moreover, in Loveman, the method to rewrite the program is dependent on the target.

On the other hand, according to independent Claims 1, 13 and 20, even if the program part which would need to be rewritten in Loveman is not rewritten by providing the definition of intrinsic function to the compiler, the same result as the result compiled by Loveman's method can be obtained. Moreover, although the program rewritten in Loveman cannot be compiled with other targets, since the definition of the intrinsic function and the conventional program are separated in the claimed configuration, the program can be compiled with every target.

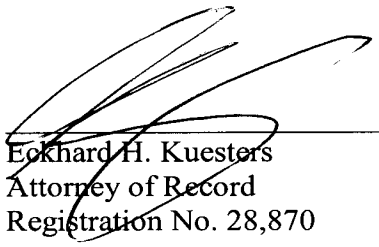
Thus, Loveman fails to teach or suggest "replacing, by a code optimizer, the multiple machine instructions by a user-defined machine instruction stored in the intrinsic function definition database in the case where the multiple machine instructions generated from the input source program by the code generator ***are determined to exactly match the machine instructions generated from the details of an intrinsic function definition body of the user-defined machine instruction***", as recited in amended independent Claims 1, 13 and 20.

Accordingly, Applicant respectfully requests that the rejection of Claims 1-20 under 35 U.S.C. §§ 102 and 103 be withdrawn.

Consequently, in light of the above discussion and in view of the present amendment, the present application is believed to be in condition for allowance. An early and favorable action to that effect is respectfully requested.

Respectfully submitted,

OBLON, SPIVAK, McCLELLAND,
MAIER & NEUSTADT, P.C.



Eckhard H. Kuesters
Attorney of Record
Registration No. 28,870

Andrew T. Harry
Registration No. 56,959

Customer Number
22850

Tel: (703) 413-3000
Fax: (703) 413 -2220
(OSMMN 08/07)

1382587_1.DOC